NATIONAL GEOSPATIAL-INTELLIGENCE AGENCY

NGA.RP.0001_1.0.0
2013-01-03

NGA RECOMMENDED PRACTICE

National System for Geospatial-Intelligence Recommended Practice for Universally Unique Identifiers (2013-01-03)

Version 1.0.0

NATIONAL CENTER FOR GEOSPATIAL INTELLIGENCE STANDARDS

Executive Summary

In order to effectively manage and use the Intelligence Community (IC) and Department of Defense (DoD) enterprises, it is essential that the content and services can be uniquely identified. While a number of identification schemes are currently in use, none have been designed to universally uniquely identify everything on the enterprise. The ITU-T X.667 Universally Unique Identifier (UUID) has filled this role on the Internet for over 25 years. This Recommended Practice describes how this proven technology should be leveraged for use in the National System of Geospatial-Intelligence (NSG)

Change Log

Description	Date	Version
Initial Version	January 03, 2013	1.0.0

Table of Contents

Table of Figures

Table of Tables

NSG Recommended Practice – Universally Unique Identifiers

1 Introduction

Identifiers have been a part of Geospatial Intelligence (GeoINT) trade-craft since the days of wet film and Mylar. They have done a good job fulfilling the purposes for which they were designed. However, as information systems become increasingly interconnected and the information content more complex, the existing identifier schemes are insufficient.

The first issue with existing identifiers is that they are not universal. Identifier schemes provide a unique identifier within the scope of that scheme. A mission ID is unique within the set of all mission IDs. A platform ID is unique within the set of platform IDs. None of these identifier schemes are capable of uniquely identifying everything, everywhere in the NSG. To manage heterogeneous resources on an enterprise level, an identifier which is guaranteed unique regardless of the creator, purpose, type, location or content of the identified entity is required.

The second issue with existing identifiers is that they were developed for processing by humans. They are constructed of alpha-numeric characters, have variable lengths and are composed of conceptual modules (date, time, sensor type, etc.). This is fine for human processing but does not work well with software. Take the typical desktop computer for example. When a user passes a file name to a program, the first thing the system does is look up the file ID (inode for you UNIX types) associated with that file name. From that point on, all processing is performed using the file ID. The file name only comes into play when the system has to communicate with a person. Given the expanding role of automated processing in support of the GeoINT mission, a similar machine processable identifier is needed for the National System for Geospatial-Intelligence.

NSG resources cannot be reliably discovered, exchanged or managed without unambiguous, content agnostic, machine processable identification. This paper lays out an identification strategy for the NSG and describes how that strategy integrates with the Intelligence Community (IC) Cloud.

2 Scope

This document defines the National System of Geospatial Intelligence Recommended Practice (RP) for the use of Universally Unique Identifiers (UUID). The NSG standard for UUID is ITU-T X.667 which is (will be) mandated in the Department of Defense (DoD) Information Technology Standards Registry (DISR) and the Intelligence Community (IC) Enterprise Registry and Repository (ER2) registries. This RP promotes a common approach to applying the UUID standard within the NSG.

It is not the purpose of UUID to replace existing identifiers. Rather, it is to provide an enterprise level means to uniquely identify any NSG resource. The UUID compliments existing identifiers. It does not replace them.

3 Normative References

The following referenced documents are indispensable for the application of this document.

ITU-T Recommendation X.667, *Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier Components*, Aug 2008
http://www.itu.int/rec/T-REC-X.667-200808-I/en

ISO/IEC 9834-8:2008, *Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier Components*, Aug 2008
http://www.iso.org/iso/catalogue_detail_ics.htm?csnumber=53416

IETF RFC 4122, *A Universally Unique IDentifier (UUID) URN Namespace*, July 2005
http://tools.ietf.org/html/rfc4122

4 Definitions

identifier
linguistically independent sequence of characters capable of uniquely and permanently identifying that with which it is associated [adapted from ISO/IEC 11179-3]

5 Abbreviations

ABI	Activity Based Intelligence
CMSTT	Common Metadata Standards Tiger Team
DCE	Distributed Computing Environment
DISR	DoD Information Technology Standards Registry
DoD	Department of Defense
ER2	IC Enterprise Registry and Repository
GeoINT	Geospatial-Intelligence
GUIDE	Globally Unique Identifier for Everything
IC	Intelligence Community
ID	Identifier
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union (ITU) Telecommunication Standardization Sector
MAC	Media Access Control
NGA	National Geospatial-Intelligence Agency
NSG	National System of Geospatial-Intelligence
OCIO	Office of the Chief Information Officer
ODNI	Office of the Director of National Intelligence
RDF	Resource Description Framework
RP	Recommended Practice
SHA	Secure Hash Algorithm
URI	Uniform Resource Identifiers
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
W3C	World Wide Web Consortium

6 Requirements

The Office of the Director of National Intelligence (ODNI) Common Metadata Standards Tiger Team (CMSTT) has identified three requirements that a universally unique resource identifier must meet. These three requirements are:

1) The identifier shall be unique and long lived
2) The identifier shall Mission Neutral / Agnostic
3) The identifier shall be representable and processable

6.1 Unique and long lived

The requirement for uniqueness is obvious. An identifier which is not unique to the resource it identifies is of little use. This requirement also precludes re-use of identifiers. It does not, however, preclude multiple identifiers for a resource.

The association between an identifier and the resource it identifies must not change over time. This is essential if we are to maintain the coherence of the enterprise body of knowledge. An identifier which can no-longer be resolved is lost knowledge.

6.2 Mission Neutral / Agnostic

Identifiers are a common element used to support all missions. To succeed in this role, the identifier format cannot make any assumptions about the mission or environment where it will be used.

Corollary 1: Identifiers shall always be unclassified

A corollary to this requirement is that an identifier must always be unclassified. Identifiers are part of that small set of metadata needed to make access control decisions. Since they may be accessed prior to access control being enforced, they must always be unclassified.

Corollary 2: Identifiers shall not convey any information about the resource

If the identifier is always unclassified, then the identifier cannot convey information about the resource. Including resource information in the identifier exposes information that can be exploited by an adversary. A population of identifiers built using the Media Access Control (MAC) address, for example, would expose the nodes on the network and the nature of the resources hosted on each.

6.3 Representable and Processable

Identifiers will be frequently created and processed. Most of this will be done by software, out of sight of users. Therefore, identifiers must be easy for systems to generate, store, transfer and process. As a consequence, identifiers will not be human friendly. NSG policy will not preclude the creation of human-friendly identifiers, but they will always serve a secondary role.

7 Standards

These requirements shall be met within the NSG using the UUID form defined in ITU-T Rec X.667 (2008). The uniqueness and persistence of the UUID is demonstrated by over 25 years

of pervasive use in almost every category of software. UUIDs are encoded as 128 bit random integers. As such, they make minimal assumptions about how they will be generated and used. Finally, UUIDs address requirement 3 in that they are easy to generate (less than 60 lines of "C" code), are represented in a machine friendly binary value, and do not require access to an external service.

X.667 specifies five versions of the UUID. Of these five, only versions 1, 4 and 5 shall be used in the NSG[1]. The X.667 encoding rules allow these three versions to be used simultaneously without conflict.

UUID Version 1 is a simple parsing of the MAC address and system clock into a 128 bit value. This is the easiest form to generate and has a very low probability of collision. However, Version 1 does not meet requirement 2 so it should not be used directly.

UUID Version 4 populates the UUID with a random number. Most random number generators are not in-fact random. If started with the same seed value, they will always generate the same sequence of numbers. Therefore, Version 4 should only be used if there is a true random number source available.

UUID Version 5 is a Secure Hash Algorithm (SHA) hash of a unique value. This version is a good answer to requirement 2. Its weakness is that the uniqueness of the input value is assumed. If the input is not unique, the output will not be either.

A good combination approach is to generate a version 1 UUID which is then used as the input value to generate a version 5 UUID. The resulting identifier complies with all three requirements.

UUID is also defined in IETF RFC 4122 and ISO/IEC 9834-8:2008. These standards are virtually identical to X.667. The ITU standard is cited because it is the only one of the three that is a publicly available approved standard. However, the content of the three is so similar that compliance with any one shall be considered compliance with all.

8 Application

8.1 Legacy Identifiers

Identifiers are not new to the NSG. Mission identifiers, platform tail numbers, tactical IDs, etc. have been around for years. They have all served their purpose and there is no reason they cannot continue to do so. However, they do not meet the requirements for universally unique identifier. The National Geospatial-Intelligence Agency (NGA) Office of the Chief Information Officer (OCIO) is pursuing a strategy where a resource may have multiple identifiers. Each identifier will be qualified as to the identifier type, so that a UUID and a Platform Tail Number can easily co-exist in the same data set. Changes to existing product specifications and standards will be worked so as to minimize the impact of existing systems.

[1] Version 2 is specific to the DCE, Version 3 was replaced by version 5

8.2 IC Cloud

The Intelligence Community (IC) is building out a Cloud computing environment to better integrate analysis and reduce duplication of IT infrastructure. The identifier standard for this environment is the Globally Unique IDentifiers for Everything (GUIDE) identifier. A GUIDE identifier has three parts:

1) Scheme: The scheme serves to identify this identifier as a GUIDE identifier
2) Prefix: The GUIDE architecture provides a federated resolver service which translates a GUIDE identifier into a Uniform Resource Locator (URL) to the resource. The Prefix specifies which resolver manages the resource. Prefix values are assigned by the GUIDE Prefix Governing Process. Each prefix value maps to one and only one Resolver.
3) Suffix: A unique value defined by the local community. For NSG systems this will be the X.667 UUID.

A populated GUIDE identifier would look like this:

GUIDE://142/550e8400-e29b-41d4-a716-446655440000

Where:

Scheme = GUIDE

Prefix = 142

Suffix = 550e8400-e29b-41d4-a716-446655440000

Resources which are assigned a GUIDE identifier are not required to be registered in a resolver. We anticipate that a “not registered” prefix will be defined.

GUIDE identifiers are simple to build. The software first generates a UUID (a library to do this is included in most software development environments), pre-pend the Prefix, then pre-pent the Scheme. There is no requirement to access an external service or to get any form of external validation. In the event that the GUIDE identifier is registered with a resolver, the identifier will be validated for format and uniqueness at that time.

8.3 When to Use a UUID

The UUID provides a globally unique, machine readable, persistent identifier for a data entity. A UUID should be applied to any data entity which is managed and accessed as a discrete unit of information. For example, a geospatial feature provides sufficient information to be useful as a unit of data for management and exploitation. A geospatial feature attribute, however, has little information value outside of the context of the feature which it describes. Therefore, a feature should have a UUID but the identifier is optional for the attribute. Table 1 lists some of the data entities found in the NSG and the UUID requirements for each.

Data Entity	Mandatory / Optional
Association Type	Mandatory. An Association is a type of Feature.
Association Role	Preferred. A UUID may not be necessary if the implementing technology maintains a clear and unambiguous linkage between the Association Type and the corresponding Association Roles in the associated Features.
Attribute	Not required but should be present if quality or history is maintained for this element. The UUID helps to preserve the alignment between the attribute and its associated metadata.
Coverage	Mandatory
Data Set	Mandatory
Feature	Mandatory
Image	Mandatory
Model	Mandatory
Non-Geographic Data Set	Mandatory
Series	Mandatory
Tile	Mandatory when the Tile is managed independent of an image.
Stereomate	Mandatory

Table 1 – Data Entity UUID Requirements

While primarily intended for data entities, the UUID should also be employed to identify non-data resources. This is particularly true when the resource is commonly describes through data entities and the association of resource and descriptive data must be unambiguous. Table 2 describes some of these non-data resources and how the UUID should be applied.

Data Entity	Mandatory / Optional
Service	Mandatory
Platform	Mandatory. May be generated from the human readable platform identifier using UUID Version 5 (SHA)
Sensor	Mandatory. May be generated from the human readable sensor identifier using UUID Version 5 (SHA)

Table 2 - Non-Data Resource UUID Requirements

8.4 UUIDs and Change

NSG resources are not static. The rules for applying a persistent identifier are less clear when the resource being identified is changing. We can help address this issue by classifying resources into one of three categories:

- Dynamic: change is a part of the very nature of these resources. The UUID persists through change.
- Mutable: these resources change periodically. They should be accompanied with a change history which allows users to determine the state and content of the resource at any time in its history. The UUID persists through change.
- Immutable: these resources never change. They can only be destroyed and replaced with a new resource. The UUID does not persist through change.

The question of when a resource changes vs. when it becomes a new resource is largely a function of the mission requirements that resource supports. Ideally all resources would be

dynamic or mutable. However, the additional metadata requirements may not be justified in light of the mission requirements.

As an example, consider a highway (Interstate 95). A highway is a Complex Feature composed of centerline, paved surface and right-of-way Features. The highway itself can be considered immutable. While the constituent Features may change, Interstate 95 is still Interstate 95. The same may not be said for the paved surface Feature. If a lane is added to a stretch of the highway, then we have to capture that change in the history and extent. We can now uniquely identify the paved surface of Interstate 95 and query the history of all changes to that stretch of asphalt. Alternately, we can bound the temporal extent of the existing paved surface Feature and create a new paved surface Feature with a temporal extent starting when the existing Feature ends. In this case we retain less metadata but can still answer the question "how many lanes did Interstate 95 have at this date?" Both approaches are valid. The choice of which to use depends on the use patterns expected of the data.

8.5 UUIDs and the Cloud

Core to most "Cloud" implementations is a BigTable[2] distributed storage system. BigTable organizes data into tables. Each data element in a table is indexed by its row and column key. The row keys are most important. These are sorted in ascending order then indexed through a hierarchical indexing scheme. Given a row key, the corresponding data can be rapidly located by traversing the indexes to the appropriate table then doing a binary search on that table. However, this approach only works if the row keys are universally unique and sortable. This requirement poses a problem for the IC Cloud since the data is heterogeneous. It comes from many sources in many formats. A standard and reliable UUID is the one thing that will be consistent for all data. Using the UUID as a row key assures that all data in the IC Cloud is identified and can be located.

8.6 UUIDs and Activity Based Intelligence

A key part of Activity Based Intelligence (ABI) is the linking of independent data entities to identify associations. Since ABI is a Multi-INT discipline, it is essential that any data from any INT can be linked to data from any other INT. Since they all have different formats and content, there must be a common framework for data linking. The Resource Description Framework (RDF) language from the W3C[3] is a tool which is commonly used to form these linkages.

RDF describes associations in terms of simple properties and property values. However, RDF does not keep those properties and values in the RDF entity. It uses Web identifiers (Uniform Resource Identifiers URI) to refer to a resource where those properties are defined and values stored. A series of RDF statements form a graph where the individual properties are nodes and the URIs which connect them are edges.

---

[2] Chang et al., *Bigtable: A Distributed Storage System for Structured Data*, Retrieved from the Internet September 20, 2012 http://research.google.com/archive/bigtable.html

[3] World Wide Web Consortium, *RDF Primer*, Retrieved from the Internet September 20, 2012, http://www.w3.org/TR/rdf-primer/

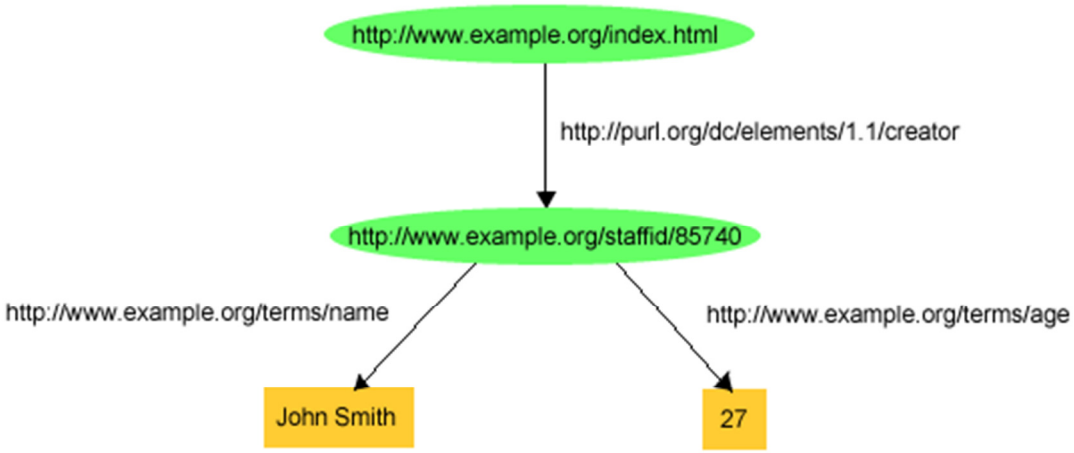

Figure 1 - Sample RDF Graph from RDF Primer

RDF is useful for ABI because it associates data by reference. There is no need for the data to be physically integrated into the graph, or even to be on the same network. However, it does require that the URI be unique and unambiguous. The UUID serves that purpose. Data tagged with a UUID is linked into an RDF graph by encoding that UUID as a URI. Resolution of the UUID would be achieved by looking up the access information though a metadata catalog or, in the IC Cloud, thought the GUIDE resolvers.